ROARING HAKING

PENGUIN SOFTWARE INC.

Vol.1 No.1
Special Issue 01/2011(1) ISSN: 1733-7186

GET RID OF SPAM!

PLUS

REVIEWS OF 2 ANTI-SPAM PROGRAMS:
HOSTED CANIT & OF CANIT ARCHIEVER

Hakin9 team

Editor in Chief: Ewa Dudzic
ewa.dudzic@*hakin9.org*

Managing Editor: *Patrycja Przybylowicz*
patrycja.przybylowicz@hakin9.org

Editorial Advisory Board: David F Skoll, Bill White, Sophia Li, Josh Audette

DTP: Ireneusz Pogroszewski
Art Director: Ireneusz Pogroszewski
ireneusz.pogroszewski@software.com.pl

Proofreaders: Michael Munt, Elliott Bujan

Top Betatesters: Aby Rao, Jum Halfpenny, Itzik Kotler

Special Thanks to the Beta testers and Proofreaders who helped us with this issue. Without their assistance there would not be a Hakin9 magazine.

Senior Consultant/Publisher: Paweł Marciniak

CEO: Ewa Dudzic
ewa.dudzic@software.com.pl

Production Director: Andrzej Kuca
andrzej.kuca@hakin9.org

Publisher: Software Press Sp. z o.o. SK
02-682 Warszawa, ul. Bokserska 1
Phone: 1 917 338 3631
www.hakin9.org/en

Dear Readers,

This issue is a result of cooperation between Hakin9 Magazine and Roaring Penguin Software Inc. All of us used to suffer and become impatient because we received spam in our private and work e-mail accounts. In this issue, we describe a solution to this problem. We also familiarize you with Roaring Penguin Software and its founder.

The introductory article give us some background about the company. The next article is an interview with David F. Skoll, the founder of Roaring Penguin. The Hakin9 team asked him about his company, product development and emerging technologies in anti-spam. Read it, and perhaps you will find the answers you were looking for. The following article – Fail Mail: How to Read a Bounce Message – is a short article written from a very practical point of view.

The article entitled Email Archiving and Continuity describes a solution for email archiving and the requirements for best archiving practice. Finally, we present 2 reviews by Hackin9 contributors of the latest Roaring Penguin Anti-Spam filtering programs.

We hope you will find this issue interesting and educational as well. Now, there is nothing more to say apart from wishing you: enjoy reading!

Hakin9 Team &
Roaring Penguin Company

ROARING PENGUIN SOFTWARE INC.

March of the Roaring Penguin

Founded about a dozen years ago as a consulting firm in Ottawa, Ontario, Canada, Roaring Penguin Software, Inc. now develops and markets several products, primarily the CanIt family of anti-spam packages for small to enterprise environments. Although the company in its current form has existed since 2002, its origins go back a little further and begin with the story of the company's founder and president, David F. Skoll.

David was born in South Africa and lived there until the age of eleven. After that, it was a change of continent and hemisphere to the rocky shores of Newfoundland, on the eastern coast of Canada. Later, he moved west to Ottawa, the capital of Canada, to work at Carleton University as a System Administrator while completing his Masters of Engineering in Electronics.

At Carleton University, David was exposed to open-source software and came to appreciate its power and flexibility. In 1994, he began using Linux and knew that Linux-based software development was his future career.

David would go on to create and donate a number of other programs that are still in use today: Remind, a sophisticated calendar and alarm program (in 1989); and RP-PPPoE, a PPPoE implementation for Linux that is deployed across Linux servers and clients worldwide (in 1998). Users of these programs have continued to get in touch with David over the years, sharing their enthusiasm. He jokes the Remind is now older than some of its users.

After obtaining his Masters degree in electrical engineering, he joined one of Ottawa's many new technology start ups with one of his professors. A couple of years later, he moved on to Chipworks Inc. where he was a Research and Development Project Leader. However, that itch to do his own thing grew. As David explains, "I had the vague idea, even then, that I wanted to have my own company. Chipworks was great and I was very happy there, but I wanted to be my own boss and had the feeling if I didn't do it now, I'd never do it." Perhaps it makes sense that someone with such a diverse background would be looking to carve his own independent and unique path.

A Tradition of Being Non-Traditional

Taking the plunge, David started his own consulting firm in 1999. He choose the name Roaring Penguin Software inspired by a cartoon in the San Jose Mercury News showing a penguin (representing Linux) biting Bill Gate's leg. The cartoon was a play on *The Mouse that Roared* and, in 1998, it was considered ridiculous to take on the giant Microsoft. David's choice is a reflection of his commitment to the Linux-based, open source philosophy / methodology whereby production and development practices promote access to the end product's source materials.

In 2000, the Royal College of Physicians and Surgeons of Canada asked David to develop an email filtering tool to stop the flood of email viruses making their way into the college's network. David set to work and created MIMEDefang as an open-source email inspection software for system administrators. Afterward, true to his open-source philosophy, David did the non-traditional thing and donated MIMEDefang to the open-source community under the GNU General Public License.

David was somewhat startled by the subsequent enthusiastic reaction to MIMEDefang from various users around the world. MIMEDefang's widespread popularity has since made it one of the most widely used Milters – and one of the best-supported open-source mail filters – in the world. Today, MIMEDefang is still maintained by Roaring Penguin Software.

REMIND

Remind is a calendar and reminder program for Linux and most UNIX Systems. David Skoll started writing Remind in 1989 because he was frustrated with the limitations of the UNIX calendar program. In the last ten years, it has accreted features and has become one of the most sophisticated calendar programs available. Some further information on Remind and a downloadable copy is available from *http://www.roaringpenguin.com/products/remind*.

By 2002, the increasing amounts of unsolicited commercial email, or *spam*, hammering businesses and users had hit a new high, and it was clear to David that something with the power of MIMEDefang was needed for people without the technical programming background to write MIMEDefang filters. The idea for CanIt was born and he set to work developing a user-friendly, commercial email anti-spam filtering software that allowed end-users to control settings using a web-based menu interface. MIMEDefang was to become the backbone of Roaring Penguin's industry-leading CanIt® antispam product line. The first corporate client was the City of Yellowknife in Canada's Northwest Territories and thay remain an active client to this day.

CanIt is designed as a server-based spam-control system built around SpamAssassin, MIMEDefang, Apache, and PostgreSQL. It features sophisticated spam-handling techniques which minimize the amount of spam the client receives while guaranteeing that an end-user will never lose a valid email. CanIt achieves extraordinarily accurate discrimination through human intervention, and includes mechanisms to minimize the amount of human intervention required.

The release and immediate positive reviews of CanIt meant that Roaring Penguin was changing from a consulting business to a full-fledged product development company. David realized this meant the next order of business was finding help. That came in the form of William (Bill) L. White, who partnered with David and became Vice President at Roaring Penguin. Bill brought with him a wealth of knowledge about technology sales and marketing backed up by twenty years of experience. The two worked together to form a common vision for the company; David focused on strategizing and developing new products while Bill went to work developing the short and long term strategies that would allow the company to thrive. As David puts it: *Bill helped transform Roaring Penguin from a one man show to what it is now, Roaring Penguin Software Inc., a software development company*.

With technology, business and marketing strategies in place, Roaring Penguin grew fast enough to require moving offices twice in one year. David and Bill also

RP-PPPoE

PPPoE (*Point-to-Point Protocol* over Ethernet) is a protocol used by many ADSL Internet Service Providers. David Skoll wrote a free PPPoE client for Linux and Solaris systems to connect to PPPoE service providers. Dubbed RP-PPPoE, this open-source product is ideal for Linux users with a DSL modem whose Internet service provider uses PPPoE. In the dozen years since its creation, RP-PPPoE has become the defacto standard for Linux PPPoE implementations. RP PPPoE is available for download from *http://www.roaringpenguin.com/products/pppoe*.

On The 'Net

- CanIt-Domain-PRO
- Hosted CanIt
- CanIt Archiver
- Hosted CanIt Archiver

made a conscious decision not to look for venture capital. They knew that they already that they had a product and that getting capital from outside sources would mean giving up a lot of control. Since CanIt was in hand and already fully developed in-house, it made more sense to simply get the product out on the market as soon as possible.

Since spam is a moving target, constantly mutating, RPS's software development team is flexible and responsive. Roaring Penguin developers are able to react very quickly to counteract new spam types. Roaring Penguin's frequently-updated data feeds of Bayes data, known phishing addresses and DNS blacklists ensure that CanIt stays up-to-date.

With the *Roaring Penguin Training Network* (RPTN is a mechanism for sharing Bayesian training), Bayes data is collected from all of RPS's clients worldwide, aggregated and then sent back to them in the daily updates. RPS also has an IP reputation data base and the IP reputations for all of the email servers seen by our clients is gathered and shared on an hourly basis.

RPS is an engineering-oriented company. Email is a mission-critical tool for most companies, so the software has to be robust. Roaring Penguin developers are obsessive about quality assurance, running thousands of unit tests and regression tests before each release.

Roaring Penguin's Technical Support staff aim to be very approachable and helpful. Not only do their support staff resolve the immediate technical problem, they also use every support call as an opportunity to proactively ask if there are any other client concerns or questions. They take the opportunity to educate and inform the client on how the software works in order to help a client choose their best configuration, to point out product functionality that may be applicable for a client's network and to discuss choices for optimal setups particular to the client in order to use the software in the most effective way possible. After all, the support staff have a vested interest in preventing future problems from occurring, and clients reap the benefit of an effective and efficient solution.

Over the last few years Roaring Penguin has expanded the CanIt family beyond basic CanIt to include;

- *CanIt-PRO*, an on-site email filtering anti-spam software solution at the server for small and medium businesses.
- *CanIt-Domain-PRO* which gives large organizations the ability to turn over spam blocking control to domain owners. Typically, it is used in multi-tenant situations such as managed service providers (MSPs), web hosts and universities.
- *Hosted CanIt* allows small businesses to benefit from managed email filtering done off-site.
- *CanIt Archiver* is a highly flexible email archiving and email continuity software designed for small- to medium-sized organizations to use on premises.
- *Hosted CanIt Archiver* is a hosted email archiving service designed for small- to medium-sized organizations who want to out-source their email archiving.

Roaring Penguin now has over five hundred clients around the world, including National Education Research Networks, Internet Service Providers, Web Hosts and Managed service Providers – all in all, a couple of million end users being protected by products from the CanIt family – and the number is growing daily.

Roaring Penguin is always looking ahead to see what's coming on the horizon. As David Skoll explains, *We're in a competitive market, but with all the buzz about cloud computing our Hosted CanIt anti-spam and Hosted CanIt Archiver (email archiving) packages are very well positioned*.

The Roaring Penguin history is a story about a philosophy of openness, and a commitment to excellence. And at the end of the day, satisfaction with a job well done.

SOPHIA LI

CANIT ARCHIVER

ARCHIVE YOUR E-MAIL

E-MAIL CONTINUITY DURING MAIL SERVER OUTAGES

POWERFUL AND EFFICIENT SEARCHING

FLEXIBLE RETENTION PERIOD FOR REGULATIONS AND POLICIES

ROARING PENGUIN SOFTWARE INC.

www.roaringpenguin.com

Interview with

David F. Skoll, Roaring Penguin Company Founder

David F. Skoll has a Bachelor of Engineering degree from Memorial University of Newfoundland and a Masters of Engineering from Carleton University in Ottawa, Canada. He has been a professional software developer since 1990 and founded Roaring Penguin in 1999.

Roaring Penguin

Roaring Penguin has primarily been in the Email filtering domain. Do you plan to diversify into other security tools and technologies?

We plan to continue our focus on email. In addition to filtering, however, we have recently introduced an email archiving and searching tool.

Do you offer any consulting services and/or training related to Email filtering?

We sell filtering products and services and we do offer training for our products, but we don't offer separate training courses apart from those related to our products and services.

Do your clients request product customization to meet their specific business need?

Yes. Some of our customers have requested different kinds of customization from user-interface changes to new filtering capabilities. As much as possible, we try to bring the customizations into the main product line. It keeps software maintenance simpler and lets all of our customers enjoy the good ideas that some customers propose.

Any cool features coming around the corner for Roaring Penguin?

We are working on making our products more elastic. In other words, in the event of a load spike, you'll be able to quickly fire up more machines, or even virtual machines on a cloud host, to handle the load. When the spike subsides, you'll be able to deprovision the machines in an orderly fashion.

Do you have any plans of moving in the Short Message Service (SMS) anti-spam domain? If so, what are some of the security challenges?

We do not have plans to do SMS filtering.

Anti-Spam

Can you list some of the emerging technologies in anti-spam?

That's hard to say because it assumes we can predict emerging technologies in spamming. Anti-spam tools tend to react to new spamming techniques.

That being said, I think some of the following technologies hold promise for the future: IP Reputation systems will become important; as mail servers share information about good and bad IP addresses, it helps to reduce the usefulness of compromised machines.

Bayesian analysis continues to be effective. It's an adaptive technology that pretty quickly learns new spammer content tricks and blocks them.

Much to my surprise, greylisting continues to be effective. Greylisting is so simple to work around that I thought it would cease to be effective years ago, but that is not the case. I guess the economics of spamming don't make it worth trying to defeat greylisting.

Email spam has existed since the advent of email technology itself. Why is it so hard to completely block spam?

The nicest, most beautiful thing about email is that anyone can quickly get in touch with anyone else, even if they don't know one another and have never communicated before. This wonderful aspect of email also enables spam. It's simply built-in and impossible to block completely. If we change email so spam becomes impossible, email won't be nearly as useful as it is now.

Email Social Engineering is on the rise. How can anti-spam mitigate social engineering? What are some of the challenges you face in anti-spam due to social engineering?

Anti-spam tools can mitigate social engineering both before and after a compromise. Before a compromise, standard anti-spam techniques can block many phishing attacks. After a compromise, outbound scanning and rate-limiting can limit the damage done by a compromised account. It can also alert administrators quickly to compromised accounts.

Social engineering is impossible to stop. The best we can do is apply technology to limit its damage and educate our users. But con artists have been around (and thrived) since antiquity and will probably continue to do so until the end of humanity.

As an anti-spam expert, what are some of the new threats you face?

We don't see any radically new threats. We see more of the same: Phishing attacks, distributed DoS attacks, fraud, viruses and plain old spam.

Besides the standard DNS blacklist and Reputation Stats what AntiSpam features do you incorporate to detect spam?

We use many techniques including greylisting, Bayesian analysis, and heuristic rulesets. End-users can make their own allow/block lists and custom rules. For outbound scanning, administrators can implement rate-limiting to catch abusive accounts.

Do you feel that AntiSpam software is fighting a losing battle?

It depends on what you mean. If the goal is to stop ALL spam without any collateral damage, then yes: That's an impossible goal. If the goal is to reduce spam to manageable levels so that email remains useful, that's a possible goal and I think we're winning that battle.

If you could alter the current Email SMTP/ Mime/MTA's..... features (SPF, DKIM, SenderId) and supported options what would you implement to help battle spam?

None at all. Any technology that blocks spam will necessarily destroy the beauty of email I mentioned earlier: The ability to dash off a message to someone you've never communicated with in the past. I think current anti-spam technology does an adequate job; there's no need to radically re-engineer SMTP.

That being said, DKIM and SPF are useful in certain restricted cases, such as to authenticate that a message purportedly from your bank really is from your bank.

That does nothing to stop spam, but it can help you trust an incoming message more.

Does your software perform any checks such as how many recipients, how many messages are being sent in one open connection? Is there throttling for unknown IPs?

Yes, all of those checks are possible, though we don't do them by default. They can be enabled easily.

What current tricks are spammers using now to evade detection? (Like text spaced out with html columns/rows..., rtl, encoding...) How do you detect them?

Obfuscation tricks come and go like changes in fashion. For a while, image spam was popular, but then it declined. We hardly see any clever obfuscation like HTML tables, fancy encoding, etc. any more.

Basically, we don't do much to detect new tricks: Our Bayesian system locks on to them pretty quickly. Occasionally, a particularly clever variant starts slipping through, so we analyze it and push out new rules to our CanIt installations to catch it.

CanIt

Can you tell us how you practice Defense in Depth in your products?

Our products run in the cloud or on gateways. We include virus scanning, but we recommend that our customers running Windows also run virus scanners on their desktops to protect against non-email virus vectors.

In terms of the CanIt architecture, our scanning code runs as an unprivileged user. A compromise of the scanning code or Web interface should not lead to a compromise of the entire machine.

We have privileged operations running in separate processes that communicate via IPC with other processes to compartmentalize high-privilege code.

What are some of the security controls put in place to protect Roaring Penguin Training Network (RPTN)?

All submissions to RPTN are encrypted using GPG and our public key. For every submission, we know the customer that submitted it. We do nightly quality-assurance checks on RPTN data and block customers who submit bad data (that is, data that disagrees substantially with other submissions).

For download, we sign the RPTN database with our GPG key. CanIt installations download the database via HTTPS and reject it if the GPG validation fails.

There are so many compliance requirements (HIPAA, SOX, OECD) regarding email archiving, how do you keep up with the requirements?

We offer basic tools for archiving (how long to keep mail, what to keep, etc.) We also ensure that our archive is immutable: End-users and CanIt administrators cannot delete or modify archived data. After that, it's up to the customer to choose settings appropriate for the regulations he or she operates under.

Hosted CanIt offers Perimeter Defense which eliminates the threat of denial of service and directory harvesting. Can you give us some technical insight into this?

CanIt cannot eliminate the threat of Denial of Service attacks, because nothing can. However, it can reduce their impact in some cases by blocking machines or senders quickly before expensive content-scanning is performed. We also have an emergency "Joe-Job" protection mechanism that blocks failure notifications if a spammer fakes spam from a victim domain.

As far as directory harvesting goes, CanIt can temporarily firewall off any machine that sends to too many invalid recipients in a given amount of time. This lets you quickly detect and shut down access to machines trying a directory harvest attack.

Product Development

It is often recommended that security management should be an integral part of software development. Do you believe in this approach?

Yes, of course. Any time you design software or write code, you need to keep in mind security threats. Bugs can easily turn into security flaws. Also, when you write software, you shouldn't only think about how it's meant to be used. You should go wild and imagine how it could possibly be abused.

What kind of software development life cycle (Waterfall mode, Agile etc) do you practice and how is security integrated in it?

We don't have a buzzword-compliant label for our software development process. We tend to write short design documents that are fairly high-level and then get right into coding. We write unit-tests as part of the code development cycle, so I guess *Agile* would be the closest buzzword for describing what we do.

You seem to have a dedicated security testing team, what security process have you adopted to ensure each of your products is thoroughly tested?

Unfortunately, there's no magic bullet. We write lots of unit tests and regression tests because bugs can become security problems. We also have pretty experienced coders who have written lots of network-facing software, so security issues are always on our minds when we write software.

Any time we find a bug or a security problem, before we fix it, we write a unit-test that demonstrates the problem. Then we fix the problem and make sure the unit-test reflects the fix.

How often do you release security updates for your products?

Any time a security problem comes to light and is fixed, we issue an update. In terms of our products specifically, there haven't been too many – maybe about 6 or 7 in the last 10 years. But we rely on many third-party products and we issue security updates any time the upstream vendor does.

Cloud Computing

Will Cloud computing play any role in anti-spam technology?

Well, sure! Cloud computing is the future of everything, isn't it? Seriously, one of the things we're working on is making our products more cloud-friendly. We want to easily deploy and decommission scanners in the cloud in response to changing load conditions.

Interview prepared by Aby Rao & Hakin9 Team

Fail Mail

BY DAVID F. SKOLL

ROARING PENGUIN SOFTWARE INC.

Email aficionados often deride postal mail as *snail mail*, but when email goes bad, the posties get their revenge: *Hah! Fail mail*

Sending email can fail for a number of reasons (for example, you could have mis-spelled the recipient's address, or the recipient's mail box could be full.) In theory, the computer that detects the failure sends you back a helpful notification explaining what went wrong. In practice, you do get back a notification – but not a very helpful one. In this article, I'll explain how to decode the dreaded *Delivery Status Notification* (DSN) also known as Delivery Failure Message.

The Nice Thing About Standards

... is that there are so many to choose from. Unfortunately, the format of failure messages varies wildly depending on the software that generates them. While there is an Internet standard for delivery status notifications (RFC 3464), it's a 40-page standard that only a committee member could love. It also doesn't produce particularly readable notifications. On the other end of the spectrum, D.J. Berntein's quirky *Qmail* program produces plain-text notifications that begin:

Hi. This is the... with a nice chatty English paragraph that might even include the text: *Sorry it didn't work out*. And yes, Qmail's notifications must begin with: *Hi. This is the...* even if they aren't in English.

For the purposes of this article, I'll concentrate on the RFC 3464-specified notifications with a brief look at Exchange-generated notifications.

User-Friendly

The RFC 3464 format is very user-friendly... if the user happens to be an unusually nerdy computer-scientist. Here's a sample:

```
The original message was received at Sat, 2 Jul 1994 17:10:28 -0400
 from root@localhost
   ----- The following addresses had delivery problems -----
 <louisl@larry.slip.umd.edu> (unrecoverable error)
 ----- Transcript of session follows -----
 <louisl@larry.slip.umd.edu>... Deferred: Connection timed out
       with larry.slip.umd.edu.
 Message could not be delivered for 5 days
 Message will be deleted from queue
```

Here are the key things to look for: The following address had delivery problems: This tells you which address failed. In this case, it's *<louisl@larry.slip.umd.edu>*.

The *Transcript of session follows* may have a few useful nuggets of information. The session here refers to the SMTP session between the machine trying to send the message and the machine that failed it. In this case, we learn that the connection to *larry.slip.umd.edu* timed out (and presumably had been timing out on each attempt for 5 days.) So the sender gave up. The report may have an additional section that looks something like this:

```
Reporting-MTA: dns; cs.utk.edu
Original-Recipient: rfc822;louisl@larry.slip.umd.edu
Final-Recipient: rfc822;louisl@larry.slip.umd.edu
Action: failed
Status: 4.0.0
Diagnostic-Code: smtp; 426 connection timed out
Last-Attempt-Date: Thu, 7 Jul 1994 17:15:49 -0400
```

The interesting bits are: **Reporting-MTA:** The machine that generated the failure notification. It is the administrator of this machine that you should pester first to find out what happened. **Original-Recipient** contains the original recipient address. (The "rfc822;" preamble just specifies the address format; in this case, a normal Internet email address.) "Final-Recipient" can be different if a user forwards his or her email; it may show the forwarding destination instead of the original recipient. **Action** shows what happened. In this case, delivery failed. Possible actions are *"failed", delayed, delivered, relayed or expanded*. (**relayed** means *I think it got through, but the machine that accepted it won't tell me for sure*. And expanded is typically seen for mailing lists; it means the original message has been forwarded, possibly to many more recipients). **Status** is a three-number field. Only the first number really matters. *2* means success, *4* means persistent temporary failure and *5* means permanent failure. The difference between *4* and *5* is subtle. A *4* means there's a problem that should eventually be fixed, but wasn't fixed in time for this message to be delivered. This could be something like a full disk or a dead machine. A *5* means it's a problem that is not likely to be fixed ever; an example would be a nonexistent recipient address.

Exchange

Microsoft, naturally, does not feel the need to abide by Internet standards. Exchange has its own non-standard failure notification that starts out "Your message did not reach some or all of the intended recipients" and gives more details. The details typically include a friendly English explanation of what happened. Unfortunately, the explanation, though friendly, is quite often wrong! That is because Microsoft in its infinite wisdom generates an explanation based on the numerical response from the other server. Luckily, recent versions of Exchange include the actual text of the error message, so you can generally ignore Microsoft's friendly version and read the real message. Here's an example:

```
The following recipient(s) could not be reached:
   sue@example.org on 4/27/2009 5:29 PM
      The message contains a content type that is not supported
      < mail.example.org #5.6.1 SMTP;
       553 5.6.1 Message subject indicates spam content.>
```

The incorrect Microsoft explanation is the unsupported content type. The correct explanation is the original error message in angle brackets.

Email Archiving and Continuity

Email archiving is becoming important for many organizations for several reasons:

- Every business has key employees or a vital business division such as finance which must keep critical records including email. Some industries such as health care are required to archive email in compliance with government regulations, e.g., HIPPA, SOX, FOIA, etc.
- Many companies archive email for purposes related to intellectual property or to facilitate discovery in the event of litigation. A searchable archive can greatly speed up the process and reduce the cost of discovery.
- A searchable archive serves as an email backup and disaster-recovery mechanism in case the primary mail server suffers failure. In this event, the archive's usefulness is dependent on its being up-to-date and how easy it is to search.
- A searchable archive allows administrators to monitor inbound and outbound messages to verify compliance with company policies.

A solution such as Roaring Penguin's CanIt Archiver can be configured to archive email for an individual or department or entire company and can satisfy all of the above requirements.

In addition, CanIt Archiver offers email continuity: If a company's primary mail server is down, employees can still read incoming messages by retrieving them from the archive. This allows timely access to time-sensitive information and ensures continued productivity even if the mail server is down.

Archiving Requirements

At a minimum, any archiving solution must preserve the following information:

- The original unmodified message. (It should not rewrite the MIME structure of the message; rather, it should preserve the message exactly as it came in over the wire.)
- All envelope information (envelope sender, envelope recipients, sending relay IP address, remote system HELO argument).
- The date and time at which the message was archived.

The archiver should permit searches based on:

- Full-text searches of the message subject and body.
- Searches based on envelope information.
- Searches based on metadata such as attachment filenames, message-IDs, relay host, etc.

The archiver should permit the end-user to view and download messages as well as resend them out of the archive. It should clearly mark resent messages with standard Resent-From and Resent-Date headers.

The archiver should allow the user to search related messages so he/she can follow the threads of back-

and-forth conversations. The archiving solution should be integrated with an anti-spam solution so it does not bother archiving spam or viruses. While it is acceptable to put the archiver after an anti-spam filter, this arrangement is less desirable because it is more difficult to accurately preserve envelope information if the archiver is not the perimeter mail relay.

Email Continuity Requirements

For email continuity, the archiver must archive and index messages quickly. Archiving should be done in real-time and the delay between archiving and indexing should be, at most, a few minutes. In order to reduce backscatter, the archiver must be capable of learning valid recipient addresses so it can continue to accept email for them even if the mail server is down, yet reject (or temporarily reject) messages for unknown recipients.

With Roaring Penguin's own CanIt Archiver, email continuity is assured as email is accessible in the event of a mail server outage and email is queued for delivery when mail service is restored.

Email Archiving Configurations

CanIt Archiver is a highly-configurable email archiving system. Email archiving may be configured for a specific end-user or group of users.

On-Premises Archiving

Figure 1 shows one arrangement for on-premises email archiving.

In Figure 1, mail arrives from the Internet and flows through the anti-spam filter and archiver to the back-end corporate mail server. Outbound mail flows the other way: from the corporate mail server through the archiver and filter and out to the Internet. Having mail flow both ways allows both inbound and outbound mail to be archived.

Archiving Internal Email

One of the problems with archiving email on a separate machine from the company mail server is that the archiver normally does not see purely internal email (because such email does not leave the corporate mail server). However, many mail servers including Sendmail and Microsoft Exchange have an option to copy or *journal* email to an external address. By using this Bcc or journalling feature, administrators can ensure the archiver captures all mail (inbound, outbound and purely internal) and has a complete record of the designated streams of user email.

CanIt Archiver supports journalling from Microsoft Exchange 2007 and 2010. It also has a milter for Sendmail and Postfix that mimics Exchange journalling

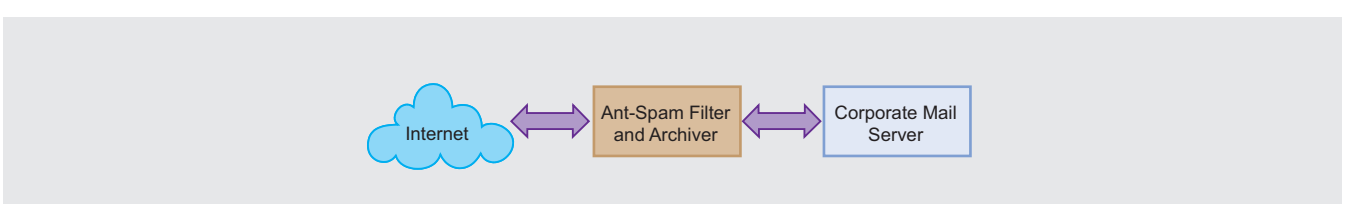


Figure 1. *On-Premises Archiving*

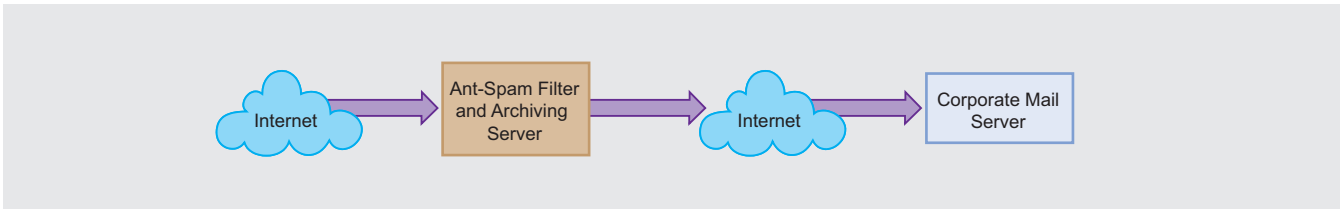


Figure 2. *Archiving as a Cloud Service*

so that Sendmail and Postfix corporate servers can archive their internal mail.

Archiving as a Cloud Service

Figure 2 shows the topology for an outsourced archiving service. Mail arrives from the Internet at the archiving service provider's machine where it is filtered and archived. It is then relayed over the Internet to the customer's back-end mail server.

Note that typically, mail does not flow out of the mail server back through the filtering and archiving service. For this reason, journalling or BCC'ing as described above is used to feed outbound and internal mail to the archiver. (Naturally, mail arriving from the cloud service need not be journalled back to it for archiving.)

Archiving Technical Issues

While archiving seems simple (*Store all the email and index it for searching*), there are a number of technical issues that complicate it.

Disk Space

Archiving messages for several years can consume a substantial amount of disk space. Consider an average-sized organization of 50 people, each of whom receives 1MB of email per day. To archive mail for three years, that organization would need about 53.5GB of disk space (plus additional overhead for the indexes).

To archive the email for 50 people for 10 years would require approximately 178 GB.

A larger organization of 5000 people would need almost 5.5TB of disk space over three years.

Backups

While 600GB or even 6TB of disk space isn't outrageous, the archive must be backed up and should be stored redundantly. An archive that can't be trusted is worse than no archive at all. Securely backing up a large archive can be time-consuming.

Encryption

The mail archive will contain a lot of sensitive information. For this reason, it should be stored on an encrypted file system and all backups should be encrypted. While an encrypted file system does not protect against an on-line attack, it does protect the archive should the physical server or a backup tape be stolen.

In addition to encrypting the archive, internal mail should be encrypted en route to the archiving machine, especially if it must traverse the Internet. This can be accomplished by using the STARTTLS extension to SMTP.

Access Control and Auditing

The archive system must ensure that users can access only mail that they would normally see (that is, only mail they have sent or received). It should also keep a complete audit trail of searches and message accesses so users and administrators can see exactly who has been searching and accessing the archive.

Archive Integrity

The system must ensure the integrity of the archive. Time stamps must be accurate, original message details must be preserved faithfully, and deletion of archived messages should be prohibited.

Reports

The archiving system should be capable of producing reports such as:

- Messages and bytes archived per day.
- Messages and bytes archived per day per customer.
- Total archive size by customer.
- Number of customer email addresses that have archived messages.

Summary

When selecting an email archiver, the system administrator is wise to consider all of the issues discussed in this white paper. A simple tool to dump masses of email into a file system may not meet the organization's needs. Choosing a dependable, easily-accessible and searchable archiving system requires more care.

DAVID F. SKOLL

David F. Skoll has a Bachelor of Engineering degree from Memorial University of Newfoundland and a Masters of Engineering from Carleton University in Ottawa, Canada. He has been a professional software developer since 1990 and founded Roaring Penguin in 1999.

CANIT DOMAIN-PRO

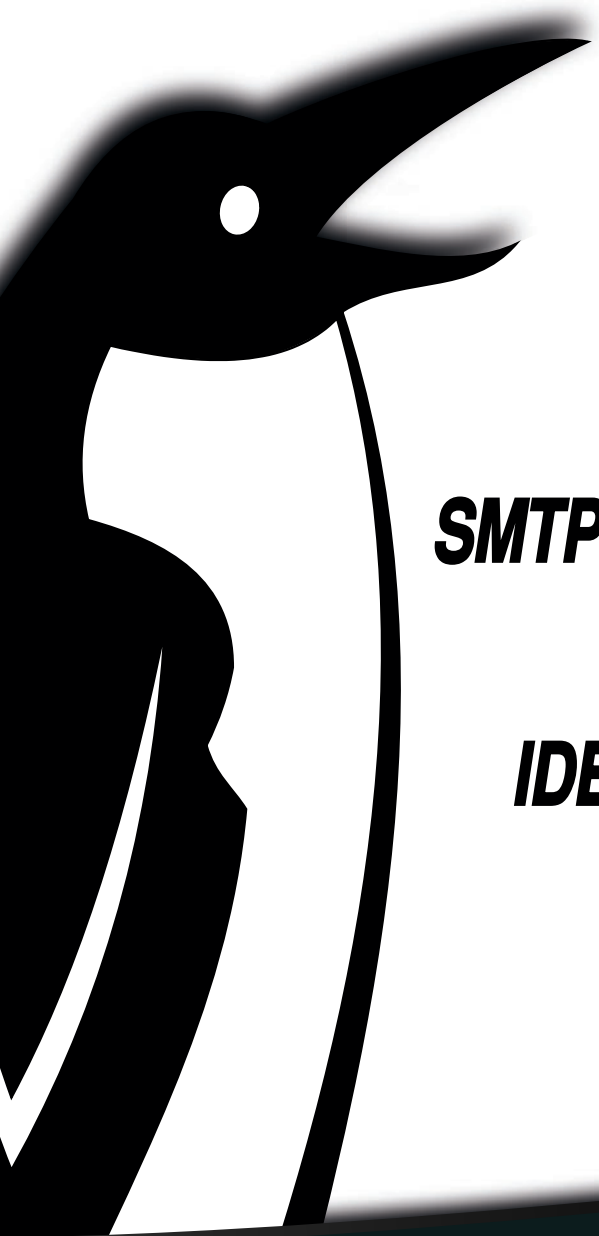
POWERFUL AND EFFECTIVE ANTI-SPAM + ANTI-VIRUS SOFTWARE

DELEGATE ADMINISTRATION TO CLIENT ADMINS AND END USERS

DEPLOYMENT OPTIONS INCLUDE CLUSTERING AND AUTOMATIC FAILOVER

SMTP COMPLIANT - WORKS WITH ALL MAIL SERVERS

IDEAL FOR MSPs, WEB HOSTS, ISPs AND UNIVERSITIES



ROARING PENGUIN SOFTWARE INC.

www.roaringpenguin.com

CanIt Archiver Review

Email is a vital tool for any modern business. The explosive growth in email volume means ensuring service levels and regulatory compliance can be a challenge.

With many public and private sector organisations mandated to retain copies of both internal and external email correspondence the burden of managing email increases. CanIt Archiver from Roaring Penguin, a Canadian anti-spam and email filtering company offers a solution. CanIt Archiver comes in two flavours. A software appliance suited for on-premise solutions and a hosted managed service. I've been looking at the latest release (version 8.0.7 at the time of writing) of their Hosted Archiver managed service.

How it Works

When you send an email your mail server will determine where it needs to delivered by looking up the *mail exchanger* (MX) records for the domain in DNS. The first step in setting up CanIt Archiver is to change the MX records for your domain to point to mail servers managed by Roaring Penguin. Effectively they receive all your email and this allows for email to be filtered and otherwise managed before it is finally sent onward to your mail server for delivery. Outgoing an internal email can also be copied to CanIt Archiver ensuring you archive your complete email correspondence.

Features

As well as providing spam filtering and service continuity should your mail server go down Roaring Penguin offers an archiving service for those who want to or need to keep a record of their email. Some organizations are legally required to retain copies of emails under regulations such as SOX, HIPAA and FOIA. This can be a headache for system administrators who manage corporate email systems that are usually not designed with archiving in mind. The volume of data to be stored can quickly mount and

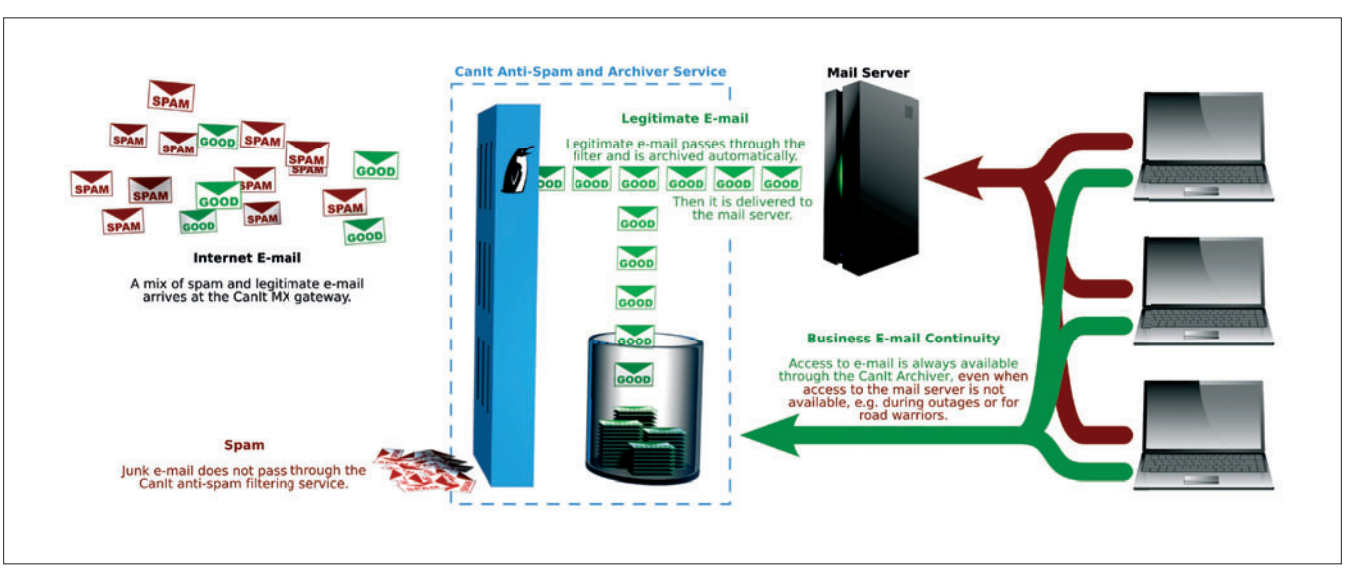


Figure 1. *Hosted Archiver sits in front of your mail server to gather, filter and archive your mail*

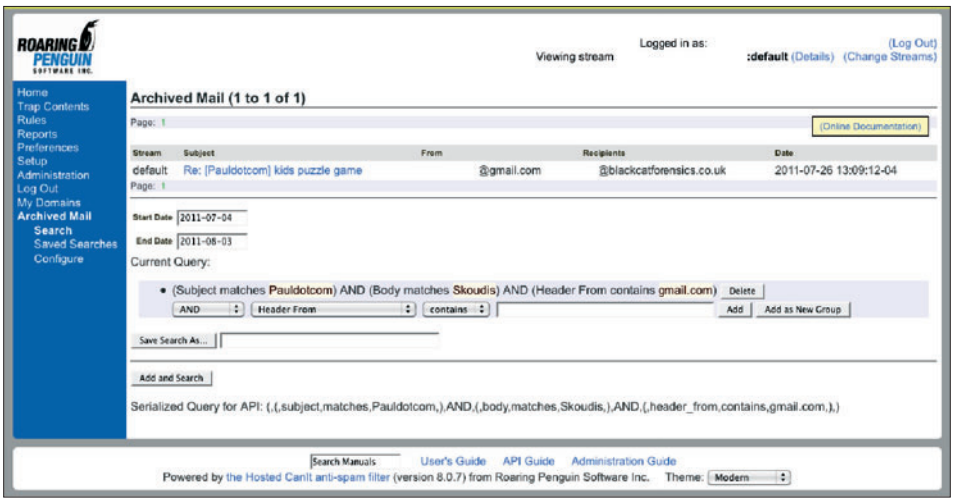


Figure 2. *Build searches in the web interface and display the equivilent API query*

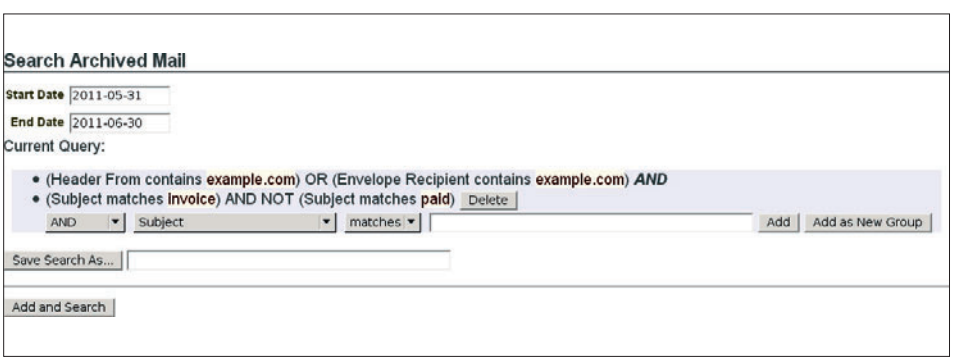


Figure 3. *Search Archived Mail Menu*

traditional backup media like tapes that are usually not lend well to rapid searching and retrieval. As the name suggests the key feature of CanIt Archiver is email archiving. CanIt Archiver automatically dedupes to save space and encrypts data to help prevent unauthorised on inadvertent disclosure. The retention period can be tuned to stay within whatever your legal mandate deems necessary.

Mining the Archive

Traditional document archives are only as good as their catalogue. You have to be able to find what your looking for after all and here's where the real power lies. One of new features in CanIt Archiver the ability to perform complex searches on your archived mail. As well as offering a web interface where you can build and execute queries, an API is available to automate and integrate queries into other management, reporting and DLP tools.

The web interface allows you to construct a search using a form and then displays the equivalent API query. This comes in handy for building and testing your API calls. Searches can be refined by adding more search terms to drill down into the results. The queries syntax is not a regular expression based so the online documentation is a a valuable resource, explaining how to build queries from scratch. Searches can be performed on email content and headers as well as other metadata including the mail relays the message passed through, the message ID or even the client HELO sent during the SMTP conversation.

References: CanIt Archiver
Email archiving solution
Website page

Mail Archive Security

Security and integrity has to be an issue for anyone looking to warehouse large volumes of email. If your entire mail spool were unsecured this would present a serious liability. In addition to encrypting the email archives on disk email in transit can be secured between the Roaring Penguin servers and your mail servers using TLS, maintaining the confidentiality of internal coresspondence. Any users with access to the archive cannot tamper with or delete emails, preserving their value as a true and accurate record of an organisation's email. Access to the archive is audited so that when someone takes a peek inside the email store a log of the transaction is recorded.

Conclusion

Hosted Archiver is a quick and simple drop-in solution to email retention, with the added benefit of being able to bolt on other services such as spam filtering. Using a managed service removes the need to plan and implement your own archival strategy in-house, although some might find the notion of storing all their email with a third party a little disconcerting. All in all Hosted Archiver is a capable product for those who simply need to tick the box on the auditor's checklist or those who want to gain deeper intelligence into their email traffic.

JIM HALFPENNY

Has been a systems administrator and consultant for over 12 years. He grew up in the era of microcomputers and cut his teeth on BBC basic and 6502 assembler. Today he work as a senior systems administrator for an ISP.

Hosted CanIt Review

Internet spam is one of the hardest forms of malicious content to stop. Spam comes in various formats, the most common of which is email spam. I have installed and evaluated the Roaring Penguin Hosted CanIt Anti Spam, a completely outsourced hosted spam filtering service.

Hosted CanIt includes detection and mitigating techniques such as keyword search, header analysis, message format analysis, Bayesian statistical analysis, blacklists, whitelists, greylists, openproxy lists, DNS verification, SpamAssassin™ content-filtering rules *sender policy framework* (SPF), custom rules and more. Its aim is to automatically protect you from spam, fraud schemes, phishing viruses and more.

Installation

The installation is pretty simple: just change your domain MX record to the Hosted CanIt Anti Spam MX server and you're all set. No software or hardware installation is required. The administrative console is web-based, and it supports themes (my favorite is Modern) that allow customizing and rebranding features.

The Administrative Console

Because email filtering is not an interactive process, I begin by describing it from the administrative console point of view. The administrative console first landing page is Home, where you can view filters and pending messages, filtering aggressiveness, and to quickly add addresses to accept/reject list. The Trap content menu allows browsing of Spam and non-Spam messages, looking up a specific incident and searching. Every incident in the messages table is hyperlinked to a new page where there is a drill down to more details and actions.

For instance, clicking on the email subject opens a page with the message content and options to do Base64 decoding, HTML tags stripping and rejection/acceptance of this message. The most interesting page

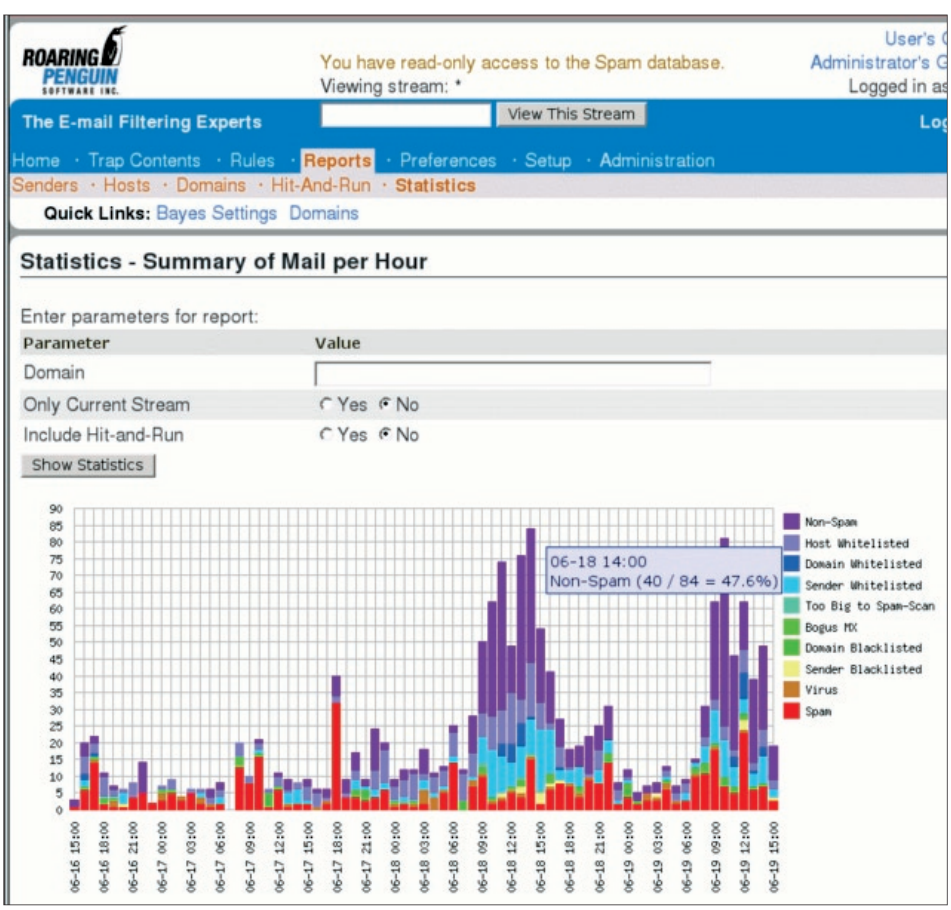


Figure 1. *Statistics – Summary of Mail per Hour*

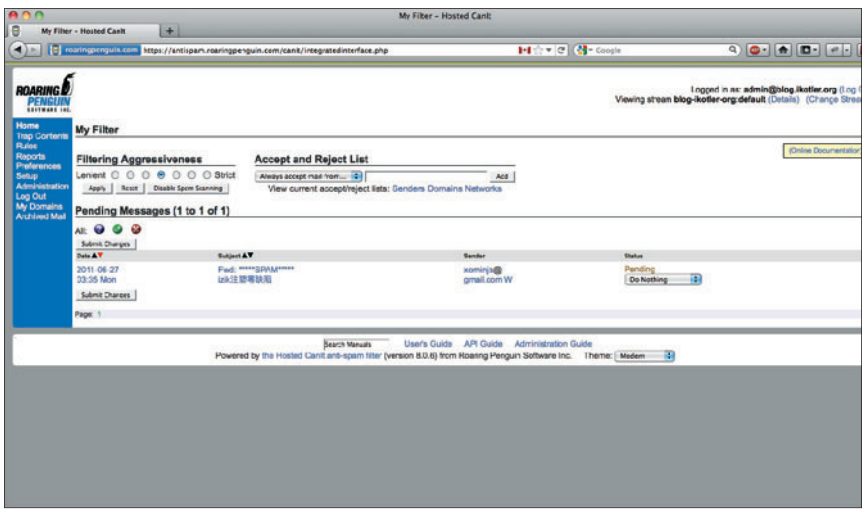


Figure 2. *Home Page Screen*

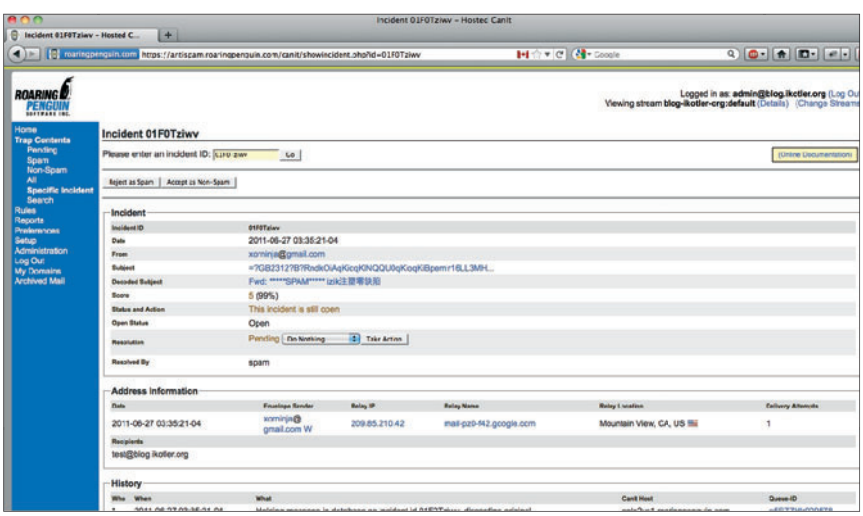


Figure 3. *Incident Report of a Given Email Screen*

for me was the incident page (reached by clicking on the message date) that included a drill down to how, why and what caused the message to be where it is. This page included information such as address information, history, and spam analysis report with mathematics behind the message spam score.

The rules menu allows you to configure specific rules as well as external sources. One interesting fact is that Roaring Penguin also maintain their own RBL rules, which of course they share with their customers.

Spam Filtering

Under the Reports tab, there are different reporting and graph options that you can generate on the fly, or ask to be generated and sent to you by email periodically. Switching to the filtering aspect of the product, in addition to the rules there is also Bayesian spam filtering. Bayesian analysis is a statistical technique that works by collecting statistics on the use of words. Without getting into details, The Bayesian system doesn't know any probabilities in advance, and must first be trained so it can build them up. To train a filter, one must indicate whether a new email is spam or not. So, to solve this problem for new customers and users, the Roaring Penguin offers a nightly-updated collection of Bayes data that is trained via customer feedback. This is in addition to personal Bayes data that you can train yourself. Another ineresting aspect of CanIt Anti-Spam is that its Bayes algorithm is programmed to also look at word pairs as opposed to just looking at words. In many cases the word pairs are the ones that stand out.

Defense Side

On the defense side, the CanIt Anti-Spam product also includes protection against dictionary attacks that are used by attackers to guess valid email addresses and is included as part of an on-going updated rules set. Phishing emails are also addressed and protected against by using customer feedback and external feeds. Now, I have conducted a non-representative test by forwarding a number of spam emails from my spam box and a number of non-spam emails and the results were to my liking. No false-positives. All in all I am quite happy in using Hosted CanIt and I like the way I am given filtering and defense in depth.

References: CanIt Archiver
Email archiving solution
Website page

ITZIK KOTLER

Itzik Kotler brings more than ten years of technical experience in the software, telecommunications and security industries. Early in his career, Itzik worked at several start-up companies as a Security Researcher. Prior to joining Security Art, Itzik worked for Radware (NASDQ: RDWR), where he managed the Security Operation Center (SOC), a vulnerability research center that develops update signatures and new techniques to defend known and undisclosed application vulnerabilities. Itzik has published several security research articles, and is a frequent speaker at industry events including Black Hat USA, RSA Europe, DEFCON, and Hack In The Box Amsterdam.

recommended by Hakin9